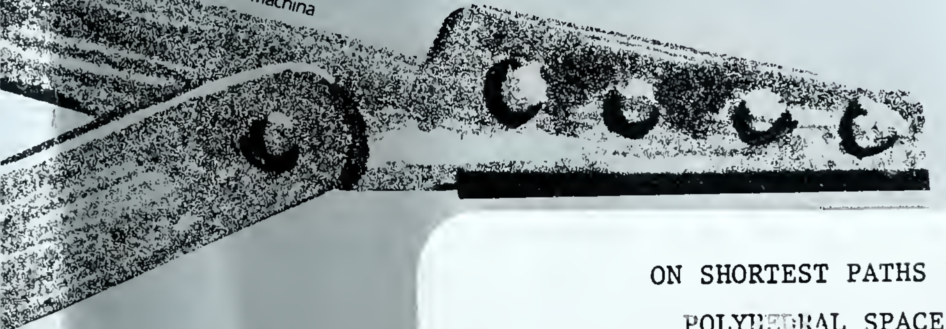
Robotics Research Technical Report

Generatorium omnis laboris ex machina



ON SHORTEST PATHS IN
POLYHEDRAL SPACES

by

Micha Sharir
and
Amir Schorr

Technical Report No. 138
Robotics Report No. 31

October, 1984

New York University
Courant Institute of Mathematical Sciences

Computer Science Division
251 Mercer Street New York, N.Y. 10012

ON SHORTEST PATHS IN

POLYHEDRAL SPACES

by

Micha Sharir
and
Amir Schorr

- - - - - - - - - - - - - - -

Technical Report No. 138
Robotics Report No. 31

October, 1984

On Shortest Paths in Polyhedral Spaces

Micha Sharir[(*)] *and Amir Schorr*

School of Mathematical Sciences

Tel Aviv University

ABSTRACT

We consider the problem of computing the shortest path between two points in two- or three-dimensional space bounded by polyhedral surfaces. In the 2-D case the problem is easily solved in time $O(n^2 \log n)$. In the general 3-D case the problem is quite hard to solve, and is not even discrete; we present a doubly-exponential procedure for solving the discrete subproblem of determining the sequence of boundary edges through which the shortest path passes. Finally we consider a favorable special case of the 3-D shortest path problem, namely that of finding the shortest path between two points along the surface of a convex polyhedron, and solve it in time $O(n^3 \log n)$.

1. Introduction

The problem of finding the shortest path between two points in Euclidean space bounded by a finite collection of polyhedral obstacles is a special case of the more general problem of planning optimal collision-free paths for a given robot system (here we treat the robot as a single moving point).

In two dimensional space the problem is easy to solve, because the shortest path between two given points must be a polygonal line whose vertices are corners of the given polygonal obstacles, so that the problem can be immediately reduced to a discrete graph searching, and can be solved in time $O(n^2 \log n)$, where n is the number of obstacle corners. This two-dimensional problem has been considered by Lozano-Perez and Wesley [LW], and later also by Lee and Preparata [LP]. In some special cases, considerably more efficient algorithms exist. For example, if the free space

(*) Work on this paper by the first author has been supported in part by a grant from the U.S.-Israeli Binational Science Foundation.

within which the shortest path is sought is the interior of a simple polygon, then the shortest path can be found in time $O(n \log n)$ [LP], [Ch]. As another example, if all the barriers are straight segments parallel to each other, then the problem can again be solved in time $O(n \log n)$ [LP] (a similar favorable case, but with a somewhat different solution, is discussed below). Another favorable case is noted by Tompa [To], where the obstacles are all convex and aligned in a certain manner along a straight line.

In three-dimensional space the problem becomes much harder. In this case the shortest path between two given points can also be shown to be a polygonal line, but all we can say about its vertices is that they lie on edges of the given polyhedral obstacles. Thus the problem is by no means discrete. Even if one knew the sequence of obstacle edges through which the desired shortest path passes, the calculation of the points of contact of the path with these edges requires solution of high-degree algebraic equations, which must be accomplished either by numerical approximate methods, or by precise, but very inefficient, symbolic algebraic calculations. Even the calculation of the sequence of obstacle edges through which the shortest path passes seems to be very difficult, and we do not know of better than doubly-exponential-time algorithms for this subproblem. Papadimitriou [Pa] has recently presented an approximating algorithm for the general three-dimensional polyhedral shortest path problem, which runs in pseudo polynomial time.

However, in certain special cases the 3-D problem is not so hard to solve. We will consider the case of finding the shortest path between two points along the surface of a convex polyhedron, a problem that has been suggested originally by H.E. Dudeney as a mathematical puzzle in 1903 (see [Ga, p. 36]; in his original formulation a spider has to crawl along the surface of a cube to reach a fly in the shortest possible manner). We present an $O(n^3 \log n)$ algorithm for this problem, which exploits the special structure of "geodesic" paths along the surface of a convex polyhedron. Our technique has later been extended by O'Rourke, Suri and Booth [OSB] to find shortest paths along the surface of a nonconvex polyhedron, and also been improved by Mount [Mo].

The paper is organized as follows. Section 2 presents a straightforward solution to the 2-D problem; Section 3 discusses the general 3-D problem, develops the relevant theory, and presents a doubly-exponential algorithm for solving the problem. Finally, Section 4 analyzes the problem of shortest paths along a convex polyhedron and an algorithm for solving this problem is presented in Section 5.

2. The Two-dimensional Case

If one assumes that the solid obstacles are all orthogonal prisms whose heights are all parallel to, say, the z-axis, and are enclosed between a floor and a ceiling, and if the two points X, Y, between which an optimal path is sought, are assumed both to have the same height, then the $3-D$ case of our problem may be reduced to the $2-D$ case, since then the optimal path will lie entirely in a horizontal plane containing X and Y, and the given solid obstacles will intersect this plane in a collection of polygonal obstacles. Let us therefore assume that V is a closed two-dimensional region bounded by a collection of polygonal walls and other polygonal obstacles, and let X, Y be two points in V. The 2-D shortest path problem is then to find a (Euclidean) shortest path between X and Y which is wholly contained in V. This problem, and several special cases of it, has already been considered by several other people ([LW], [LP], [Ch], [To]). The general approach used below has been suggested by Lozano-Perez and Wesley [LW], but without any explicit analysis of its complexity. Most of the other relevant work has involved special cases of the problem, where more efficient solutions than the one presented below exist. To simplify the following discussion, we assume that each corner of the boundary of V is incident to just two edges (although the method described below will also apply in cases where this assumption does not hold). It is easily seen that the shortest route from X to Y which is wholly contained in V is a polygonal path connecting X to Y whose intermediate corners are all vertices of the polygonal walls and obstacles bounding V. Furthermore, a vertex p will follow another vertex q in such a route only if p and q are *visible* from each other (in V), i.e. only if the straight segment joining p and q lies wholly in V. See Fig. 2.1 for illustration of the concepts just discussed.

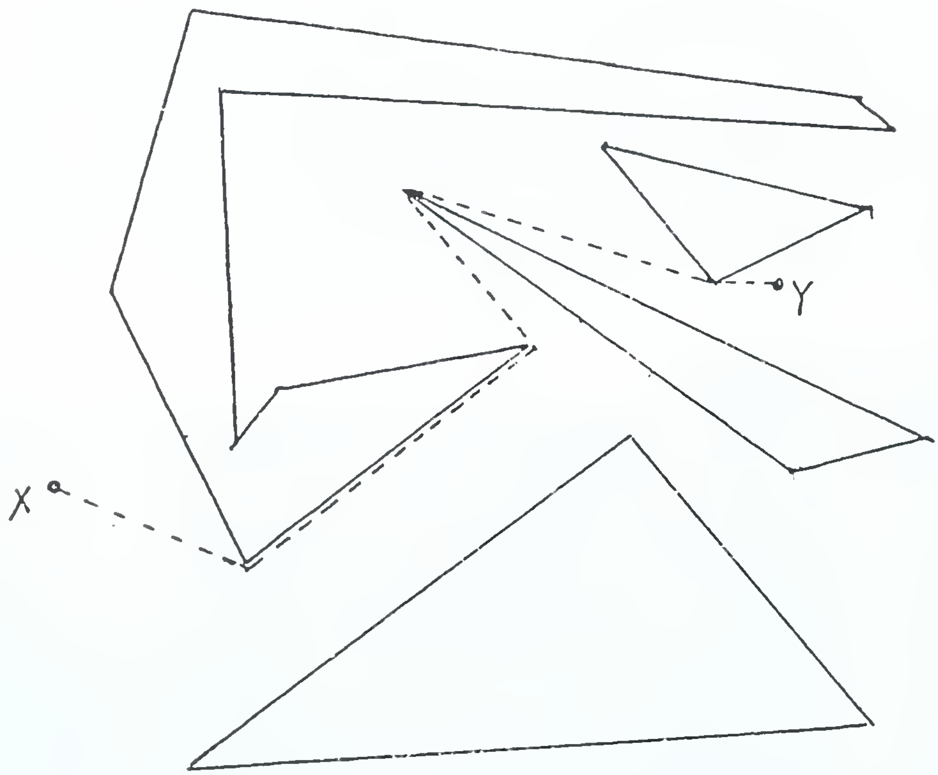


Fig. 2.1. The shortest path problem in 2-D polygonal space.

Thus, to solve the $2-D$ version of our problem, we first construct a *visibility graph VG*, whose nodes are X, Y, and the vertices of the boundary of V, and each of whose edges connects a pair of vertices visible from each other, and has length equal to the distance between these vertices. Then we search through VG to find the shortest path in VG from X to Y. This yields a polynomial-time discrete algorithm which can be implemented to run in time $O(n^2 \log n)$. For the sake of completeness, we sketch here such a straightforward implementation.

Let N denote the set containing X, Y, and all the vertices of the boundary of V ($bd(V)$ for short). For each $u \in N$, we will find all points in N visible from u by using the following algorithm which is based upon a 'plane-sweeping' technique similar to those used in [Sh], [NP]. Specifically, for each orientation θ let $l = l_\theta$ denote a ray extending from u at orientation θ. Let $M(\theta)$ denote the list of all edges in $bd(V)$ whose interior intersects l, sorted in increasing distance of these intersections from u. We maintain $M(\theta)$ as a 2-3 tree, and can easily compute an initial value of M for some starting orientation θ_0 in time $O(n \log n)$.

We next rotate the ray l in clockwise direction about u, and update the value of $M(\theta)$ each time l crosses an orientation θ at which M changes, either due to addition of new edges into M or deletion of edges from M, or both. It is clear that such changes can occur only at orientations θ of l at which l passes through another corner in N. We thus iterate through these critical orientations in clockwise order (including also the orientations of the rays connecting u to X, Y). Let θ_p be such an orientation at which l passes through $p \in N$. Then M will change at θ_p so that the edges incident to p and presently in M are deleted from M, and the other edges incident to p are inserted into M. However, if the ray l passes simultaneously through several points $p \in N$, then we have to remove from M all edges incident to any of these corners, and to add to M all other such incident edges, except for edges which lie on l, which are ignored in this updating process.

At each such orientation θ_p we also check whether the first edge in M has changed. If it did then that corner p on l incident to this edge is visible from u, and we update VG by adding to it the edge (u,p). Otherwise none of these p's is visible from u. If p is X or Y, then it will be visible from u if and only if it precedes along l the first edge in M.

Iterating in this manner through all critical orientations θ_p, we find all points in N visible from u in time $O(n \log n)$. Hence, if we repeat this procedure for each $u \in N$, we can construct the visibility graph VG in time $O(n^2 \log n)$.

Having constructed the visibility graph, we can then search through it to find the shortest path in VG connecting X and Y, using Dijkstra's algorithm (see e.g. [AHU]), which will run in time $O(n^2)$.

Improving the Efficiency of the Algorithm.

We next discuss an alternative approach to the 2-D shortest path problem which leads in some special cases to more efficient algorithms. We begin by the following observation:

Definition: For each point $Z \in V$, let $\pi(Z)$ denote a shortest path from X to Z through V.

Lemma 2.1: Let Z_1, $Z_2 \in V$. Then either $\pi(Z_1)$ and $\pi(Z_2)$ do not intersect each other or, if

they do intersect at some *last* point Z, then Z must be a corner in $bd(V)$, and the lengths of the initial portions of both paths between X and Z are equal.

Proof: If $\pi(Z_1)$ and $\pi(Z_2)$ meet at a point Z, then the lengths of their initial portions up to Z must be equal, or else we could replace the longer such initial portion by the shorter one, and so shorten the length of one of these paths. The same argument also implies that Z must be a corner, because otherwise, after replacing one initial portion by the other one, we would obtain a shortest path to one of the points Z_1, Z_2 which is polygonal and has a corner at an interior point of V, contradicting the basic properties of such shortest paths noted above. Q.E.D.

We can use this observation to obtain an $O(n \log n)$ algorithm for finding the shortest path in the following special case (which is similar to the second special case considered in [LP]): Suppose that the boundary of V consists of k vertical lines, denoted $l_1, \ldots, l_k$, each of which contains several point apertures through which one can cross from one side of the line to the other. The passage is blocked however at all other points on these barriers. Suppose further that X lies to the left of all these barriers and that Y lies to the right of all of them. Note that the shortest path from X to Y through V must pass through exactly one aperture at each of the lines $l_1, \ldots, l_k$. Hence it can be found in $O(n^2)$ time, using a standard dynamic programming approach. However, using Lemma 2.1, we can improve this procedure as follows (a similar divide-and-conquer approach has been used by Reif [Re] for a different problem involving planar networks): Suppose that the barrier l_j has n_j apertures, $j=1, \ldots, k$ (so that $\sum_{j=1}^{k} n_j = n$). We will process the barriers from left to right. For each barrier l_j we will compute for each aperture $Z \in l_j$ the length $d(Z)$ of the shortest path from X to Z through V, and also the aperture $p(Z) \in l_{j-1}$ through which the shortest path passes just before reaching Z. For each $Z \in l_1$ we put $d(Z) = |XZ|$ and $p(Z) = X$.

Suppose that these maps have already been computed for all apertures lying on l_{j-1}. Let Z be the median of all apertures along l_j. We compute $d(Z)$ and $p(Z)$ by trying to pass the path $\pi(Z)$ through each of the apertures in l_{j-1} (as in the standard dynamic programming approach). Let

$W = p(Z)$, and assume that W is the m-th highest point along l_{j-1}. Since, by Lemma 2.1, shortest paths can be assumed not to cross each other, it follows that we can partition the problem into two subproblems: First find the shortest paths leading to the highest half of the apertures along l_j, using only the highest m apertures along l_{j-1} as possible predecessors along such paths, and then repeat this procedure for the lowest half of the apertures on l_j, using this time only the lowest $n_{j-1}-m+1$ apertures along l_{j-1} as possible predecessors.

Repeating this procedure recursively, it is easily seen that it will find (correctly) the shortest paths to all apertures on l_j in time $O((n_{j-1}+n_j) \log n_j)$. Hence if we iterate in this manner through all barriers l_j, we obtain an $O(n \log n)$ algorithm for finding the desired shortest path from X to Y.

The special case just considered has led to a favorable algorithm because of the regular structure of shortest paths in this case. Other special cases have been considered in [LP], [Ch], [To]. In the general case shortest paths may behave less regularly, although they still do not intersect each other. In fact, it is easily seen that, for a given starting point X, the set A containing all the corners of $bd(V)$ and X, can be arranged in a tree T with X as the root, such that each corner u is the son of a point v if the last straight segment on $\pi(u)$ is vu. Moreover, for each $u \in A$ let $\Gamma(u)$ denote the set of all points $y \in V$ for which the last segment on $\pi(Y)$ is uy. Note that $\Gamma(u)$ is nonempty only if the angle within V between the last segment vu on $\pi(u)$ and one of the edges e of $bd(V)$ incident to u is greater than 180 degrees. In this case $\Gamma(u)$ is contained in the wedge formed between e and the straight ray continuing vu past u. It is also easy to show, by techniques similar to those used to analyze Voronoi diagrama (cf. [Sh], [FAV]) that the boundary arcs between adjacent regions $\Gamma(u)$ are all straight or hyperbolic arcs, and that there are at most $O(n)$ such arcs.

In summary, the collection of shortest paths within V from some fixed starting point X can be characterized by a combinatorial structure whose size is $O(n)$, and it therefore seems likely that faster than quadratic algorithms for its construction should exist. There exist some other special cases where this is indeed the case. For example, if V is the interior of a simple polygon,

then shortest paths within V can be computed in time $O(n \log n)$ (cf. [LP], [Ch]); another favorable case has been noted by Tompa [To] in connection with wire routing problems in VLSI. However, for general polygonal regions V the problem of computing shortest paths within V in faster than $O(n^2 \log n)$ time is still open.

3. The Three Dimensional Case

The situation becomes much more complicated when we pass to the 3-dimensional version of the problem. Here it is easy to check that the shortest path from X to Y consists of a polygonal path whose vertices (except for X and Y) lie on some of the edges of $bd(V)$. The problem therefore is not immediately seen to be discrete, since there seems to be a continuum of potential paths to be considered. We will see however that the problem can be discretized, and develop an algorithm for finding the shortest path which runs in doubly exponential time in the number of wall edges.

We will find it useful to regard the wall edges as open segments, so that the wall corners are disjoint from the edges. The collection of wall edges and corners will sometimes be refered to as "wall objects".

We will first consider the following subproblem: Given a sequence $\xi = (\xi_1, \ldots, \xi_n)$ of wall objects, find the shortest path π from X to Y constrained to pass through each of the objects $\xi_1, \ldots, \xi_n$ in this order, assuming that no other constraint is being imposed on the path. Let π consist of the segments $\pi_0, \ldots, \pi_n$.

Lemma 3.1: For each $i=1, \ldots, n$, if ξ_i is a wall edge (rather than a wall corner) then the angles that π_{i-1} and π_i subtend at ξ_i are equal.

Proof: This is well known, but to see this take the two planes formed by π_{i-1} and ξ_i and by π_i and ξ_i, and "unfold" them about ξ_i so as to make them coincident, with π_{i-1} and π_i lying on different sides of ξ_i. Since π is the shortest possible, the two segments π_{i-1} and π_i must be collinear in this common plane, and the angles which they form with ξ_i must therefore be equal. Q.E.D.

Let us temporarily assume that all wall objects $\xi_1, \ldots, \xi_n$ are segments, rather than points. An initial attempt to exploit Lemma 3.1 in finding the required shortest path π, is to determine the point Z_1 of contact of π with ξ_1. This point determines the angle at which π_0 meets ξ_1, which must be equal to the angle at which π_1 leaves ξ_1. Knowing this angle, we can determine the point(s) Z_2 at which π_1 will intersect ξ_2, and continue to proceed in this manner through all segments in ξ. A correct choice for Z_1 is one in which the final angle at which π_n must leave ξ_n from their point of contact Z_n, is equal to the angle subtended at ξ_n by Z_nY.

However, this approach is problematical in the sense that each time we try to extend the path π from a point Z_i on ξ_i to ξ_{i+1}, there can exist two points Z, Z' on this edge for which the two angles subtended at ξ_i by the segments Z_iZ and Z_iZ' are both equal to the angle subtended at ξ_i by π_{i-1}. Thus, even if we fix the first point of contact Z_1, the number of paths that may arise in the way described above may be exponential in n. To overcome this difficulty, we will make use of the following observation:

Lemma 3.2: Let l_1, l_2 be two lines in 3-space, let A, B be two distinct points on l_1, and let C, D be two points on l_2. Let the angle between the vectors $\vec{AB}$ and $\vec{AC}$ (resp. between $\vec{AB}$ and $\vec{BD}$) be α (resp. β). Similarly, let the angle between the vector $\vec{CD}$ and $\vec{AC}$ (resp. between $\vec{CD}$ and $\vec{BD}$) be γ (resp. δ). Then $\alpha > \beta$ implies that $\gamma > \delta$ (see Fig. 3.1).

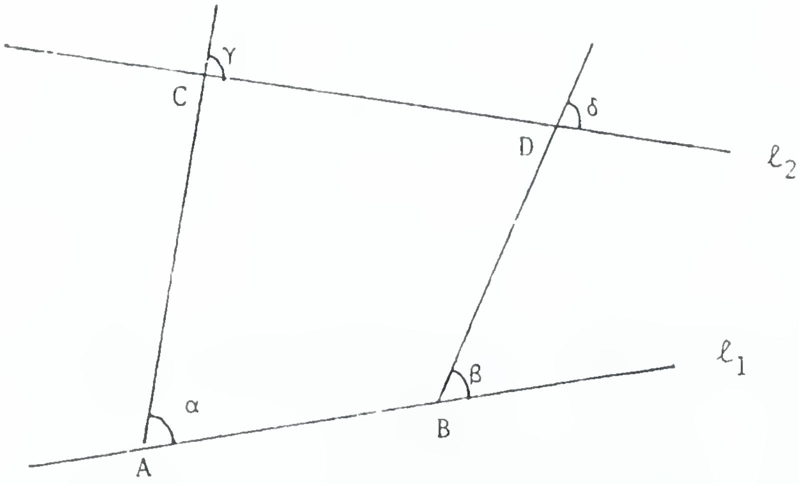


Fig. 3.1.

Proof: First note that under these conditions we must have $C \neq D$ (and thus in particular γ and δ are well defined), for otherwise α would be an interior angle in the triangle ABC which is larger than the exterior angle β, which is impossible. Put $\vec{AB} = \mathbf{u}$, $\vec{AC} = \mathbf{x}$, $\vec{BD} = \mathbf{y}$. Then $\vec{CD} = \mathbf{u}+\mathbf{y}-\mathbf{x}$. Since $\alpha>\beta$ we have

$$\frac{\mathbf{x}\cdot\mathbf{u}}{|\mathbf{x}|} \leq \frac{\mathbf{y}\cdot\mathbf{u}}{|\mathbf{y}|}$$

To prove $\gamma>\delta$ we need to show that

$$\frac{\mathbf{x}\cdot(\mathbf{u}+\mathbf{y}-\mathbf{x})}{|\mathbf{x}|} < \frac{\mathbf{y}\cdot(\mathbf{u}+\mathbf{y}-\mathbf{x})}{|\mathbf{y}|}$$

By the first inequality, it suffices to show that

$$\frac{\mathbf{x}\cdot(\mathbf{y}-\mathbf{x})}{|\mathbf{x}|} < \frac{\mathbf{y}\cdot(\mathbf{y}-\mathbf{x})}{|\mathbf{y}|}$$

or that

$$\frac{\mathbf{x}\cdot\mathbf{y}}{|\mathbf{x}|} - |\mathbf{x}| < |\mathbf{y}| - \frac{\mathbf{x}\cdot\mathbf{y}}{|\mathbf{y}|}$$

or

$$\mathbf{x}\cdot\mathbf{y} < |\mathbf{x}||\mathbf{y}|$$

which is immediate, since $\alpha \neq \beta$. Q.E.D.

Remark: The assertion in the preceding lemma can be extended to the case $A = B$ (so that α

and β are undefined). If $C \neq D$ then we always have $\gamma > \delta$ since γ is an exterior angle and δ is another interior angle in the triangle ACD.

Lemma 3.3: The shortest path π from X to Y which passes through the sequence of lines $\xi_1, \ldots, \xi_n$ in this order is unique. (Note that we assume here that $\xi_1, \ldots, \xi_n$ are full lines, or, alternatively that π passes through interior points of $\xi_1, \ldots, \xi_n$.)

Proof: Suppose that there exist two shortest paths π, π' from X to Y through the lines $\xi_1, \ldots, \xi_n$. Apply Lemma 3.2 for each of the (skew) quadrangles whose edges are ξ_i, ξ_{i+1}, π_i, π_i', $i=0, \ldots, n$, where π_i (resp. π_i') is the i-th segment along π (resp. π') (see Fig. 3.2).

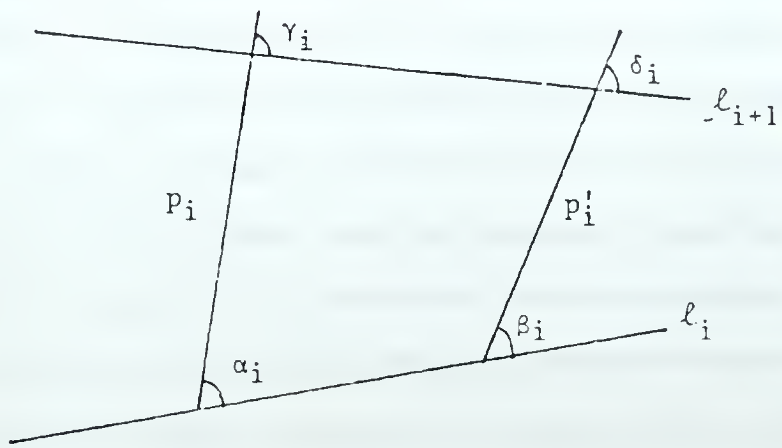


Fig. 3.2.

Lemma 3.1 implies that $\alpha_{i+1} = \gamma_i$ and $\beta_{i+1} = \delta_i$ for each $i=0, \ldots, n-1$. The first divergence of π and π' (i.e. the first j for which π_j and π_j' have the same starting point but different endpoints) forms a triangle, and we therefore have $\gamma_j > \delta_j$. Thus, inductive applications of Lemma 3.2 imply that $\alpha_i > \beta_i$ for each $i=j+1, \ldots, n$. In particular, $\alpha_n > \beta_n$, which is impossible since π_n and π_n' meet at Y, forming a second triangle (which might degenerate to a single segment), a contradiction which proves the lemma. Q.E.D.

Lemma 3.3 can be strengthened as follows: Call a path π from X to Y which passes through

the lines $\xi_1, \ldots, \xi_n$ *geodesic* (or *locally shortest*) if, for each $i=1, \ldots, n$, the path π enters and leaves ξ_i at equal angles. It is easily checked that every path whose length is a local extremum (as a function of its points of contact with $\xi_1, \ldots, \xi_n$) is geodesic. It is plain that Lemma 3.3 remains true if one assumes that the paths in question are only geodesic. Hence we have

Corollary 3.4: There exists a unique geodesic path π from X to Y which passes through a given sequence of lines $\xi_1, \ldots, \xi_n$.

In other words, the length of the path (as a function of the contact points) has one global minimum, and no other local extrema. Thus, if we continue to assume that $\xi_1, \ldots, \xi_n$ are full lines, then we can use two different techniques for the calculation of the contact points of the required shortest path with these lines. The first technique uses approximate numerical methods for finding the required minimum. For example, we can initially pass a path π^0 through an arbitrary sequence of points, one on each of the given lines. Then, iteratively, improve the path by replacing each contact point at which the incoming and outgoing angles are not equal by another point on the same line at which these angles become equal (without changing the other contact points). An explicit formula for finding the new point of contact can be readily obtained, using elementary vector techniques. (The problem involved here is, given two points $\mathbf{a}$ and $\mathbf{b}$ outside a given line l, to find a point $\mathbf{x}$ on l such that both vectors $\mathbf{x} - \mathbf{a}$ and $\mathbf{b} - \mathbf{x}$ form the same angle with l. We leave it to the reader to verify that this condition can be expressed by an equation which is quadratic in the coordinates of each of the points $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{x}$, and whose overall degree is 4.) Each such iterative step shortens the length of the path, and the sequence of paths thus obtained will converge to a path of locally extremal length, and hence to the desired shortest path, in virtue of Corollary 3.4.

If we wish to avoid numerical analysis, and insist on obtaining a precise solution using only symbolic calculations, then we can write down a system of n quartic equations in the n positions of the contact points of the path with the given lines, each such equation corresponding to one of the constraints given by Lemma 3.1, namely that at each line ξ the incoming and outgoing angles

subtended by the shortest path be equal. This system can then be solved by elimination techniques (cf. [Wa]), leading to a single polynomial equation $p(x_1) = 0$ in, say, the position x_1 of the first point of contact. By Corollary 3.4, this equation will have a unique real solution which can then be rationally approximated to any desired degree of accuracy.

Note however that the resulting polynomial $p(x)$ will in general be of degree which is doubly exponential in n, because each elimination step computes resultants of polynomials, and thus results in polynomials of one less variable but of degree which is roughly the square of the degree of the previous polynomials, and because $n-1$ such elimination steps are required to eliminate all but one of the n variables in question. This indicates that the problem of computing the points of contacts (and thus also the length) of a geodesic path with a sequence of line segments is probably intractable. To be more precise, suppose that we have two candidate sequences ξ and η of lines, and we wish to determine whether the geodesic path passing through ξ is longer than the geodesic path passing through η. This can be solved precisely by standard methods involving symbolic calculations (as reviewed e.g. in [SS]; see also below), but these methods, which use space decomposition techniques closely related to the elimination method just noted, would also require doubly exponential time. We have not been able to prove that this problem is intractable. However, it is well known that testing similar properties of real roots of a system of low degree polynomial equations in n variables (or even determining whether such roots exist) is NP-complete (this is shown e.g. by direct reduction from 3-SAT).

Another technical problem that should be noted is that if $\xi_1, \ldots, \xi_n$ are only segments and not full lines, the global minimum may be attained at points lying outside those segments. In this case it is clear that the shortest path π from X to Y constrained to pass through these segments in order will have to pass through some endpoints of these segments, at which it will generally form unequal incoming and outgoing angles. If this is the case, and if the endpoints through which π must pass are known, then the path-finding problem reduces to a collection of subproblems, each of which calls for the computation of the shortest path between some pair of points, which is constrained to pass through a specified sequence of lines. The solution of each

such subproblem can be obtained by the methods outlined above.

Since in general the shortest path from X to Y will be a concatenation of subpaths, each connecting a pair of points (each of which is either X, Y, or a nonconvex wall corner) and constrained to pass through a sequence of wall edges, we will consider the path-finding problem as essentially solved (up to numerical or symbolic calculations of the sort discussed above) if we can specify the sequence of wall edges and corners through which the desired shortest path must pass. In this setting the problem is reduced to a purely combinatorial one, which we will refer to as the *combinatorial shortest path* problem. It is noteworthy that this combinatorial problem is at least solvable in finite time:

Proposition 3.5: The combinatorial shortest path problem is solvable in doubly exponential time by precise, symbolic calculations. If one is allowed to use numerical analysis techniques, then the problem can be solved in $O(n^n)$ steps, each step consisting of finding a shortest path constrained to pass through some sequence of wall edges and corners.

Proof: Note first that it suffices to consider only a finite number of possible sequences of wall edges and corners through which the shortest path from X to Y can pass, because the shortest path will not pass twice through the same wall corner or edge. This makes it plain that the number of such sequences that need be considered is at most $O(n^n)$. For each pair of such sequences ξ, η, apply the procedure outlined above which uses symbolic computations, to find whether the shortest path constrained to pass through the elements of ξ is shorter than the path constrained to pass through the elements of η. The sequence for which the length of the corresponding path is smallest is then the solution to our combinatorial problem. Note that the basic step that this method employs is comparison between two algebraic numbers, each of which is specified in terms of the unique real roots $r_1, \ldots, r_k$ of some system of k polynomials in k variables, for some $k \le n$ (these roots are the points of contact of one of the shortest paths with the wall edges through which it is constrained to pass). Such a comparison can be performed in precise terms using Collins' cylindrical algebraic decomposition technique for analyzing semi-algebraic sets defined by a system of p polynomial equalities and inequalities in k variables, having maximum

degree m ([Co]; see also [SS]). Collins' technique runs in $O((mp)^{3^k})$ time. In our case each polynomial equation is quartic, and p, $k \le n$. It follows that the overall cost of the combinatorial shortest path problem is doubly exponential in n. When numerical methods are allowed in the evaluation of each of the shortest paths constrained to pass through some sequence of wall edges and corners, the problem can plainly be solved in $O(n^n)$ such numerical evaluations. Q.E.D.

It is currently an open problem whether faster procedures than the straightforward one just sketched exist for solving the combinatorial shortest path problem.

4. Shortest Paths Along a Convex Polyhedron

In this section we analyze the problem of calculating the shortest path between two points along the surface of a convex polyhedron in 3-space. This special case is favorable because of various properties of geodesic paths along a convex polyhedron. These properties will be analyzed in this section; an algorithm for the calculation of such shortest paths will be presented in the following section.

Let K be a given convex polyhedron, and let S denote its boundary. Let X and Y be two points on S. The problem that we consider is to calculate the shortest path from X to Y constrained to lie along S. It will be more convenient to consider a somewhat more general problem, namely - given a point X on S, we wish to pre-process K so that later, for any desired destination point Y on S, the shortest path from X to Y can be easily and quickly calculated. To simplify the foregoing analysis we will assume, without real loss of generality, that the representation of K is nondegenerate, in the sense that no two faces of K are coplanar. (Otherwise, we can repeatedly combine pairs of adjacent co-planar faces into single faces, until the above property holds for K.) Let n be the number of vertices of K, so that K has also $O(n)$ edges and faces.

Definition: (a) A point $Z \in S$ is called a *ridge* point if there exist at least two shortest paths from X to Z along S (cf. Fig. 4.1.) We denote by R the set of all ridge points in S.

(b) For each point $Z \in S - R$, let $\pi(Z)$ denote the unique shortest path from X to Z along S.

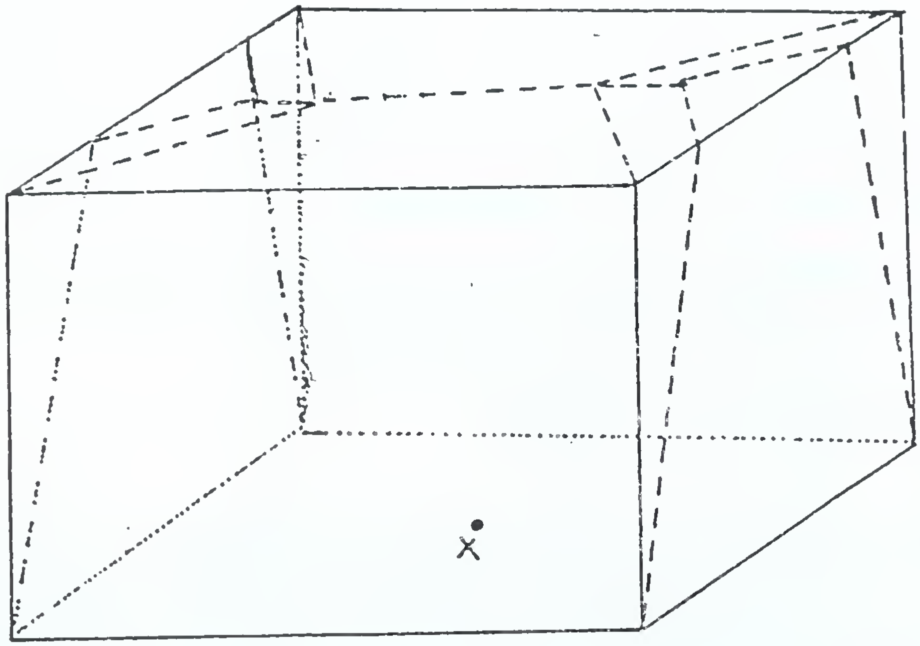


Fig. 4.1. Ridge points along a convex polyhedron

We will first consider the following subproblem: Let $Z \in S$, and suppose that the sequence $\xi = (\xi_1, \ldots, \xi_n)$ of edges of K through which $\pi(Z)$ passes has already been found (we will shortly see in Lemma 4.1 that $\pi(Z)$ cannot cross a vertex of K, so that it meets each of the edges $\xi_1, \ldots, \xi_n$ in an interior point of that edge). Then we wish to find the points of contact between $\pi(Z)$ and each of the edges ξ_i. In the general 3-dimensional case treated in the previous section, the solution of this subproblem had been rather complicated, and involved solution of high-degree algebraic equations, mainly because any two adjacent edges ξ_{i-1} and ξ_i in ξ could be skew to one another. However, in the special case which concerns us here this cannot happen, thereby making the problem immediately solvable as follows.

Let π consist of the segments $\pi_0, \ldots, \pi_n$. Recall that Lemma 3.1 implies that for each $i=1, \ldots, n$, the angles that π_{i-1} and π_i subtend at ξ_i are equal. This suggests the following simple algorithm for the calculation of the points of contact. Since we know the sequence of edges through which $\pi(Z)$ passes, we also know the corresponding sequence $f_0, f_1, \ldots, f_n$ of faces of K through which $\pi(Z)$ passes, where the face f_i contains the two edges ξ_i and ξ_{i+1}, for

$i = 1, \ldots, n-1$, and where f_0 contains X and ξ_1 and f_n contains ξ_n and Z.

We then unfold the collection of faces $f_0, \ldots, f_n$ so as to make them all lie in the same plane L. This is done iteratively. That is, initially we place f_0 in L, letting X coincide with the origin. Suppose that we have already unfolded and placed in L all faces up to f_{i-1}. We then unfold f_i about ξ_i until it becomes co-planar with f_{i-1} (but lies on the other side of ξ_i). In practice, we compute for each face f_i the displacement a_i and orientation θ_i defining its position in L relative to some standard and fixed plane representation of this face. We can then compute from a_n and θ_n the position of Z in L. The required path $\pi(Z)$, unfolded to L, is then simply the straight segment XZ. The points of intersection of this segment with the unfolded edges $\xi_1, \ldots, \xi_n$ are then readily determined, and can be easily transformed back to the original polyhedron. For further reference, let us call this process as the *planar unfolding of K relative to* $\xi_1, \ldots, \xi_n$; we also refer to the pair (a_n, θ_n) as the position of f_n in that planar unfolding. (See also Alexandrov [Al] for an analysis of the unfolded planar structure of K; the above observations have also been made by Franklin et al. [FAV], [FA] although they have not developed them into a polynomial-time algorithm).

Hence, as in the general 3-D case, the main problem which needs to be solved is that of calculating the sequence $\xi_1, \ldots, \xi_n$ of edges through which the path $\pi(Y)$ passes.

To this end, we will partition S into at most n vertex-free connected regions, called ***peels***, such that the interiors of these regions do not contain any ridge point, and such that for each such region p, the path $\pi(Z)$ to any $Z \in p$ is wholly contained in p. Since p is vertex-free, the sequence of edges through which $\pi(Z)$ passes will be easy to calculate, as will be shown below.

To obtain this partitioning, we begin with analysis of several properties of ridge points.

Lemma 4.1: A shortest path $\pi(Z)$ cannot pass through a vertex of K.

Proof: (We are indebted to R. Pollack for suggesting this simplified proof.) Suppose the contrary, and let U be a vertex of K lying on a shortest path $\pi(Z)$ from X to some $Z \in S$. Suppose first that K is incident to exactly three faces of K. Let A and B be two points on $\pi(Z)$ lying on the two straight subsegments of $\pi(Z)$ adjacent to U, with A lying before U and B after U along that

path. Let f_A, f_B be the faces containing A and B respectively, and let f be the third face of K containing U. Instead of the planar unfolding of K in which $\pi(Z)$ is a straight segment (call this the "straight" unfolding of $\pi(Z)$), we construct another unfolding as follows. First unfold faces of K into the plane as in the planar unfolding of $\pi(Z)$, until U is reached. Then, instead of unfolding f_B past f_A (as is done in the straight unfolding of $\pi(Z)$), unfold first f past f_A along their common edge, and then unfold f_B past f along their common edge; then continue to unfold as in the remainder of the straight unfolding of $\pi(Z)$ (see Fig. 4.2).

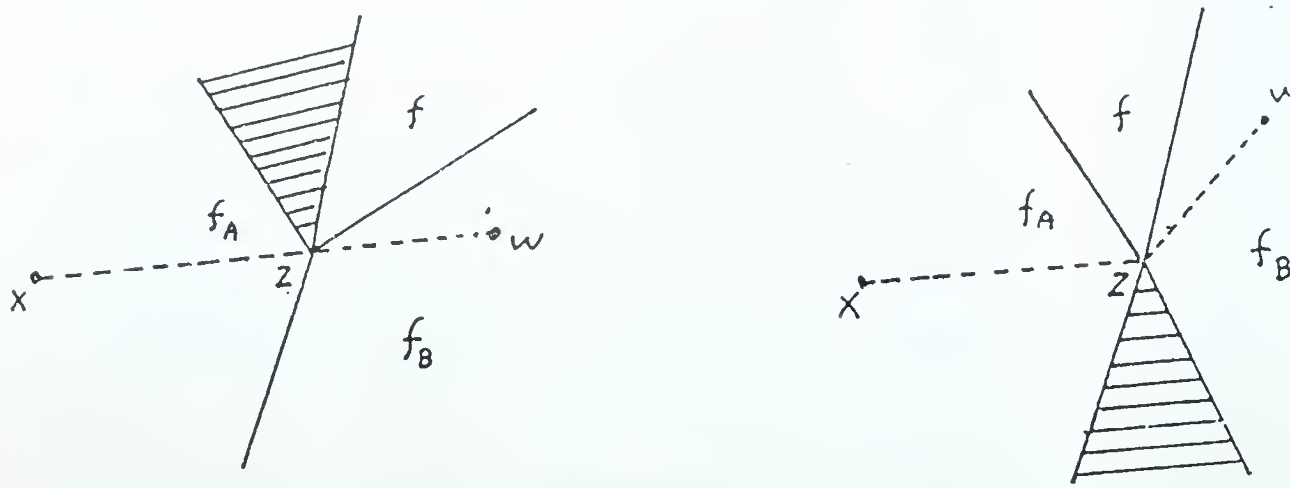


Fig. 4.2. Proof of Lemma 4.1.

In this new unfolding the path $\pi(Z)$ appears as a broken line at U, and it is easily seen that $\pi(Z)$ can be shortcut near U, yielding a shorter path which is also contained within this new unfolding. This contradiction establishes the lemma. Essentially the same argument also applies in case U is incident to more than three faces. Q.E.D.

Lemma 4.2: A shortest path $\pi(Z)$ cannot pass through a ridge point.

Proof: Suppose that $\pi(Z)$ does pass through a ridge point W. Then the initial portion π_1 of $\pi(Z)$ up to W is one of several shortest paths from X to W. Let π_2 be another such shortest path, and without loss of generality assume that π_1 and π_2 are transversal to one another at W

(this will be the case if we choose W to be the first ridge point along $\pi(Z)$, which is always well defined, because the set or ridge points along $\pi(Z)$ is closed, as can be easily verified). Let π' denote the path obtained by replacing π_1 by π_2 in $\pi(Z)$; note that π' is also a shortest path to Z. However, if W is interior to some face of K, then π' cannot be a shortest path to W, since it bends at an interior point of a face of K. On the other hand, if W lies on an edge of K, then π' cannot be a shortest path to W since it forms unequal angles with the edge containing W (note that W cannot be a vertex by Lemma 4.1). Q.E.D.

Lemma 4.3: The set R of ridge points is the union of finitely many straight segments.

Proof: With any point Z of R we can associate the face of K containing Z, and the two sequences of edges of K through which the two shortest paths from X to Z pass. Since there are only finitely many values that each of these three parameters can assume (because a shortest path to a point cannot pass through an edge of K more than once), it suffices to show that the locus of all ridge points Z lying on a fixed face of K, for which the two shortest paths from X to Z pass through two fixed sequences of edges of K, is a straight segment.

Let therefore f be a fixed face of K, and let $\xi = (\xi_1, \ldots, \xi_n)$, $\eta = (\eta_1, \ldots, \eta_m)$ be two fixed sequences of edges of K, such that any two adjacent edges in either sequence lie on a common face, and such that ξ_1 and η_1 bound the face containing X, while ξ_n and η_m bound f. Assume that there exists at least one point $Z \in R$ having these values as its associated parameters. Let (a_ξ, θ_ξ), (a_η, θ_η) be the positions of f in the planar unfoldings of K relative to the two sequences ξ and η respectively, where a_ξ and a_η both give the position of Z in the corresponding planar unfolding. Let $W \in R$ be another point having the same parameters as Z, and write $ZW = \mathbf{w}$ in the standard Cartesian representation of f. Then we have (where R_ξ (resp. R_η) denotes the rotation of the plane by the angle θ_ξ (resp. θ_η)):

$$|a_\xi + R_\xi \mathbf{w}| = |a_\eta + R_\eta \mathbf{w}|$$

which states the equality of the lengths of the two shortest paths from X to W, constrained to pass respectively through the sequences ξ and η of edges of K. Squaring out the above equation, and using the fact that $|R_\xi \mathbf{w}| = |R_\eta \mathbf{w}|$, and that $|a_\xi| = |a_\eta|$, we obtain

$$a_\xi \cdot R_\xi \mathbf{w} = a_\eta \cdot R_\eta \mathbf{w}$$

which is the equation of a straight line passing through Z. Q.E.D.

Remarks: (1) The preceding equation is nondegenerate provided that $a_\xi R_\xi \neq a_\eta R_\eta$. But if these two quantities were equal, then the two corresponding shortest paths to W would be such that their terminal segments along f coincide. This however implies that these two paths pass through another ridge point (e.g. any interior point of these coincident segments), which is impossible, by Lemma 4.2.

(2) The equation defining the ridge segment given above can be rewritten as

$$(a_\xi R_\xi - a_\eta R_\eta) \cdot \mathbf{w} = 0$$

and thus be given the following geometric interpretation:

Perform the two planar unfoldings relative to ξ and η. First rotate each unfolding in clockwise direction by the respective angle θ_ξ, θ_η (note that these rotations cause the final face f in each of the unfoldings to have the same orientation as the standard representation of f). Next translate (without rotating) the two resulting copies of f so that they both coincide with the standard representation of f. Note that this will have moved the starting point X to the planar positions $X_\xi = a_\xi R_\xi$ and $X_\eta = a_\eta R_\eta$ respectively. Then the required ridge segment (in the standard representation of f) is contained in the perpendicular bisector to the segment $X_\xi X_\eta$ (see Fig. 4.3).

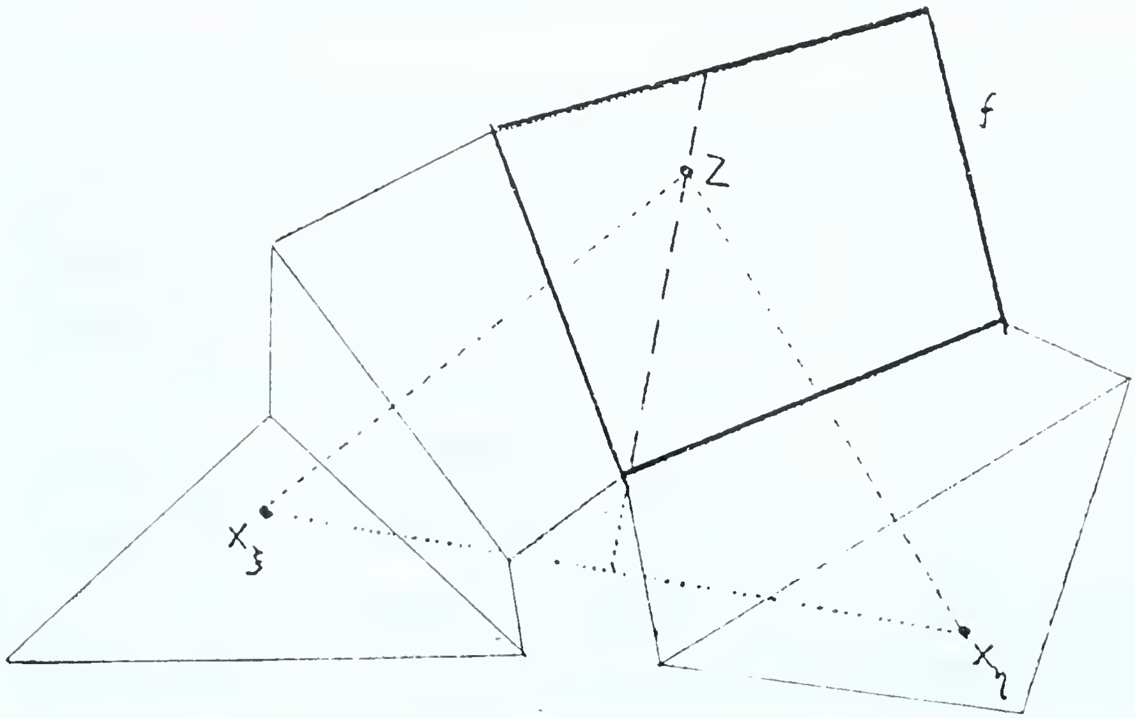


Fig. 4.3. Construction of ridge segments

(3) This construction of ridge segments makes it clear that if Z lies on a ridge segment e defined by two edge sequences ξ and η and Z is not a vertex of K, then for points W lying on the line containing e in a sufficiently small neighborhood of Z the following properties hold: There exist two geodesic (but not necessarily shortest) paths to W having equal lengths and passing respectively through the sequences ξ and η; moreover the starting orientation of each of these paths is a monotone and continuous function of W. (Note that these geodesics will be shortest paths for W lying on at least one side of Z.)

To continue our analysis of the structure of R, we first introduce the following notation.

Definition: (a) For each planar orientation θ define a polygonal path $p \equiv p(\theta)$ from X along S as follows: p starts at X in the direction of θ (relative to the standard representation of the face f_0 containing X). Whenever p reaches an edge e of K it bends over it to the adjacent face so that the two segments of p adjacent to e form with it equal angles (as in Lemma 3.1). p will terminate as soons as it reaches a vertex of K.

(b) For each orientation θ and each $Z \in p(\theta)$ we define $p(\theta, Z_\theta)$ to be the initial portion of $p(\theta)$ between X and Z.

With some exceptions noted below, we will consider the path $p(\theta)$ to terminate after the first time it reaches either a vertex of K or a ridge point. (Note that this must occur for each θ, because otherwise either the length of p would increase without bound, while p still being the shortest path to any of its points, which is plainly impossible, or else p would reach X again, in which case it is clear that p contains a ridge point.) Let $r(\theta)$ denote the endpoint of $p(\theta)$.

Lemma 4.4: (a) Let Z_n be a sequence of points on S converging to some $Z \in S$ as $n \to \infty$. For each $n \geq 1$ let π_n be a shortest path from X to Z_n, and suppose that the paths π_n converge in the Hausdorff topology of sets to a path π. Then π is a shortest path to Z.

(b) The function $r(\theta)$ is continuous.

Proof: (a) Suppose the contrary, and let π' be a path from X to Z which is shorter than π. Since the length of π is the limit of the lengths of the paths π_n, it follows that if n is sufficiently large, by appending to π' a short path connecting Z to Z_n, we can obtain a path to Z_n which is shorter than π_n, a contradiction.

(b) Let $\theta_n \to \theta$, and suppose that $Z_n = r(\theta_n)$ converges to some point Z, and that for each n the point $r(\theta_n)$ is a ridge point (if the latter property cannot be achieved, then from a certain n on, $r(\theta_n)$ is a vertex, which implies that the sequence θ_n is constant too, in which case there is nothing to prove). It follows that for each n there exists another orientation ϕ_n such that $r(\theta_n) = r(\phi_n) = Z_n$. Passing to subsequences if necessary, we can assume that ϕ_n converge to some ϕ, and that the paths $p(\theta_n, Z_n)$ converge to $p(\theta, Z)$ and similarly the paths $p(\phi_n, Z_n)$ converge to $p(\phi, Z)$. Let ξ^n, η^n be the two sequences of edges of K through which the two paths $p(\theta_n, Z_n)$ and $p(\phi_n, Z_n)$ pass. Passing again to a subsequence if necessary, we can assume that the sequences ξ^n and η^n are both constant (necessarily distinct from one another). Two cases can arise:

(i) If $\theta \neq \phi$ then by (a) the two distinct paths $p(\theta, Z)$ and $p(\phi, Z)$ are shortest paths to Z, so

that Z is a ridge point, and $r(\theta) = Z$, because $p(\theta,Z)$ does not pass through a vertex of K or a ridge point (other than Z).

(ii) If $\theta = \phi$ then Z must be a vertex of K. Indeed, if this were not the case, then (by Lemma 4.1) there would exist a sufficiently small neighborhood U of $p(\theta,Z)$ which contains no vertex of K. But then for n sufficiently large the two paths $p(\theta_n,Z_n)$, $p(\phi_n,Z_n)$ would be wholly contained in U, and thus will cross the same sequence of edges of K, contrary to assumption. The same argument as in (i) above now implies that $Z = r(\theta)$.

This proves part (b) of the lemma. Q.E.D.

Corollary: The set R^* consisting of all vertices of K and ridge points is a closed connected set.

Proof: It suffices to show that the map r defined above is onto R^*, for then R^* will be the continuous image of the unit circle of starting orientations θ. To show that r is onto, let $Z \in R^*$, and let $\pi(Z)$ be one of the shortest paths to Z. Then, arguing as in the preceding proof, $Z = r(\theta)$, where θ is the starting orientation of $\pi(Z)$ (as follows from Lemmas 4.1 and 4.2 and from the definition of r). Q.E.D.

In view of Lemma 4.3, the set R^* can be regarded as a graph whose edges are (portions of) the straight segments yielded by the proof of the preceding Lemma. The vertices of this graph are either vertices of K, or points at which such a segment intersects an edge of K, or points at which two such segments intersect. The *degree* of each vertex u of R^* is defined to be the number of edges of R^* incident to u. More information on the structure of the vertices of R is obtained from the following lemmas.

Lemma 4.5: (a) Each vertex of R^* having degree 1 is a vertex of K.

(b) Each vertex of R^* having degree 2 is an intersection of R with the interior of some edge of K.

(c) R^* does not contain any closed path.

Proof: (a) Suppose that Z is a vertex of R^* of degree 1 which is not a vertex of K. Then Z must be a ridge point. Suppose first that there are exactly two shortest paths π and π' reaching Z. By

Lemma 4.3 Z lies on a segment e which is determined by the two sequences ξ, η of edges of K through which π and π' respectively pass. If Z is an endpoint of e (and of no other ridge segment) then for each point W on the line containing e lying near Z on the other side of e there must exist a unique shortest path $\pi(W)$ to W. However, this path passes through a sequence ζ of edges of K which is distinct from both ξ and η. For suppose that $\pi(W)$ passes through the sequence ξ; then by definition of e there would exist another path from X to W having the same length as $\pi(W)$ and passing through the sequence η of edges (cf. Remark (3) following Lemma 4.3). Hence, letting $W \to Z$ along this line, we would obtain a third shortest path to Z passing through the sequence ζ, contrary to assumption. This argument can be generalized to the case in which more than two shortest paths reach Z, thus proving (a).

(b) Suppose the contrary, and let $Z \in R$ be an interior point of some face f of K, which is a common endpoint of exactly two ridge segments e_1 and e_2. Let ξ^i, η^i be two distinct pairs of sequences of edges of K which define e_i, $i=1,2$. Suppose for definiteness that $\xi^1 \neq \xi^2$. Then there are at least three shortest paths reaching Z and passing respectively through the sequences ξ^1, ξ^2 and η^1 (see Fig. 4.4). Consider the perpendicular bisector l to the segment $X_{\xi^1}X_{\xi^2}$ as in the ridge construction procedure described above. l is distinct from the lines containing e_1 and e_2, it passes through Z, and contains no ridge point in a small neighborhood of Z, except Z itself. Thus for each $W \in l$ near Z there exists a unique shortest path $\pi(W)$ passing through a sequence $\zeta(W)$ of edges of K which is distinct from ξ^i, η^i, $i=1,2$, as can be seen from Remark (3) following Lemma 4.3. But then, using arguments similar to those used in the preceding paragraph, we obtain a contradiction which concludes the proof of (b).

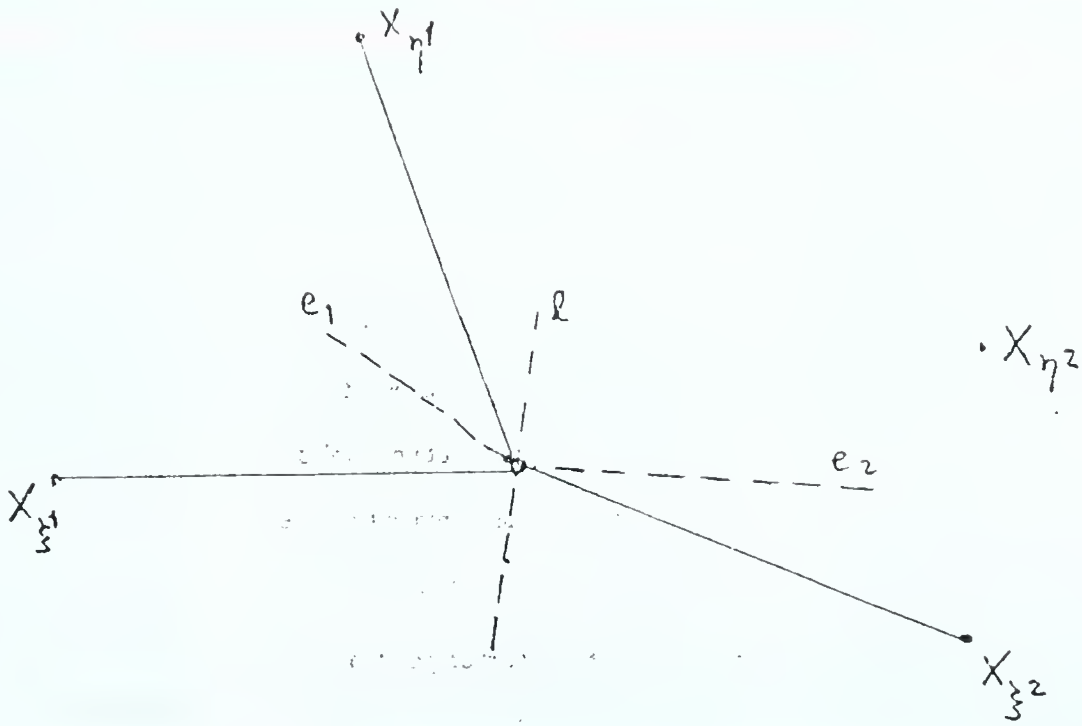


Fig. 4.4. Illustrating the proof that ridge segments cannot "bend" at a point interior to a face.

(c) Suppose the contrary, and let C be a simple closed path in R^*. By the Jordan curve theorem, C divides S into two disjoint regions S_1, S_2, so that one of them, say S_1 contains X. Let Z be an arbitrary point in S_2. Then the shortest path $\pi(Z)$ from X to Z will have to intersect C, which contradicts either Lemma 4.1 or Lemma 4.2. Q.E.D.

Corollary: R^* is a tree having (some of) the vertices of K as leaves.

Proof: Immediate by the preceding lemma.

Suppose that we have managed to construct R^*. We can then use it for calculating shortest paths from X to arbitrary points $Y \in S$ as follows:

Define Q to be the union of R with the shortest paths from X to every vertex of K. Q partitions S into disjoint connected regions (which we call *peels*), the interiors of which do not contain any vertex or ridge point. Let p be one of these peels. Using arguments similar to the preceding ones, we can show that the (unique) shortest path $\pi(Z)$ to any interior point Z of p is wholly contained in p. Also, there exists an open interval $I = I(p)$ of orientations such that

$p = \bigcup_{\theta \in I} p(\theta, r(\theta))$, and such that for each $\theta \in I$, the endpoint $r(\theta)$ does not reach a vertex. Moreover, the two end orientations θ_1 and θ_2 of I are such that both $p(\theta_1)$ and $p(\theta_2)$ terminate at a vertex of K (otherwise I and p could be increased with all the other properties of p still valid). Finally, the intervals $I(p)$ are pairwise disjoint, and the union of their closures cover the whole angular space $[0,2\pi]$.

Let J_p denote the set of edges of K which intersect p. It follows that the edges J_p can be ordered by their adjacency in p. That is, we say that e precede e' in J_p if there exists a shortest path in p passing first through e and later through e'. To see that this order is well defined, and to gain more insight into the structure of peels, we have the following lemmas:

Lemma 4.6: Suppose that a peel p is unfolded into the plane by unfolding each of the geodesic paths $p(\theta)$ in p into the ray from X at orientation θ. Then the resulting image of p is convex.

Proof: Let P be the unfolded image of p. P is plainly contained in the angular sector consisting of rays at orientations $\theta \in (\theta_1, \theta_2)$ and is star shaped with respect to X. Hence if P is not convex there must exist a concave corner Z on its boundary but not on either of the two rays $\theta = \theta_1$ or $\theta = \theta_2$. Hence Z is a ridge point. Let the two (ridge) edges meeting at Z be e_1 and e_2, and let l be the line containing e_1. Then points $W \in l$ near Z but on the other side of e_1 will be inside P, so that the shortest path to such a W is unique, is contained in p, and passes through the same sequence of edges as the path which reaches Z from within p. But then Remark (3) following the construction of ridge segments given above implies that there also exists another geodesic path to W whose length is equal to the length of the geodesic path to W through p. This contradiction proves the assertion. Q.E.D.

Lemma 4.7: Let p be a peel and let f be a face of K such that $p \cap f \neq \emptyset$. Then there exists a unique sequence ξ of edges of K such that for each $Z \in p \cap f$ the shortest path $\pi(Z)$ to Z passes through the sequence ξ.

Proof: Suppose the contrary, and let Z_1, Z_2 be two points in $p \cap f$ for which the sequences ξ, η of edges of K through which the paths $\pi(Z_1)$, $\pi(Z_2)$ respectively pass are distinct. By Lemma

4.6 the segment Z_1Z_2 is contained in $p \cap f$, and by continuity there would have to exist a ridge point $Z \in Z_1Z_2$ (hence in p), a contradiction which proves our claim. Q.E.D.

Hence each peel p defines at most $O(n)$ distinct sequences of edges of K, one for each face of K which p intersects, such that the shortest path to any $Z \in p$ passes through one of these sequences. These sequences can be arranged in an auxiliary tree associated with p so that ξ is an ancestor of η in this tree if and only if ξ is a prefix of η. Given a peel p, it is straightforward to compute all these sequences, and we omit details of this easy construction.

These observations lead to the following

Proposition 4.8: (a) There are n peels.

(b) There are $O(n^2)$ edges in R^*.

Proof: (a) Immediate from the preceding considerations.

(b) We will show that each face of K contains $O(n)$ segments of R^*. Let f be a face of K, and let $\xi^1, \ldots, \xi^t$ denote the collection of all sequences of edges of K traversed by shortest paths from X to points on f. Each such sequence corresponds to a unique peel, so that, by Lemma 4.7, $t \leq n$. For each such sequence ξ^i let $(a_{\xi^i}, \theta_{\xi^i})$ be the parameters describing the position of f in the planar unfolding corresponding to the sequence ξ^i, and let $X_{\xi^i} = a_{\xi^i} R_{\theta_{\xi^i}}$ denote the planar position of the point X when this unfolding is moved so that f coincides with its standard planar representation.

In a manner quite similar to the analysis of Voronoi diagrams [Sh] we can define the *dual* graph of $R^* \cap f$ to consist of the points X_{ξ^i} as its nodes, such that X_{ξ^i} and X_{ξ^j} are connected by an edge if the two sequences ξ^i and ξ^j define an edge of $R^* \cap f$. As in [Sh], one can show that this dual graph is planar by embedding it into the plane as follows. Map each node X_{ξ^i} to its plane position, and map each edge (X_{ξ^i}, X_{ξ^j}) to the union of two segments connecting respectively X_{ξ^i}, X_{ξ^j} to a point on the common edge of $R^* \cap f$ which these two sequences define. Since this graph is planar, and since Lemma 4.5(b) implies that it has no multiple edges, it follows that it has at most $O(n)$ edges, thus proving (b). (A similar connection between the partitioning of S by the set

of ridge points and Voronoi diagrams is also noted in [FAV].) Q.E.D.

Remark: The slicing of the surface S of K into peels has the following geometric implication. If we apply the planar unfolding procedure to all the peels, from the same planar position of X, we obtain a planar layout which we denote as $U(K)$ which has the following properties:

(1) No two unfolded peels overlap in the plane (except at points lying on an unfolded geodesic path connecting X to a vertex of K). Indeed, if Z_1 is a point in one peel and Z_2 is a point in another, then the segments XZ_1 and XZ_2 have different orientations, unless the exceptional condition noted above holds.

(2) $U(K)$ is star shaped with respect to X.

(3) If $V(K)$ is another planar unfolding of the whole surface of K having properties (1) and (2), then the smallest disc about X containing $U(K)$ has radius smaller than or equal to that of the corresponding such disc containing $V(K)$.

The partitioning of S into peels by Q enables us to find the shortest path from X to any $Y \in S$ in the following simple manner:

(a) Find the peel p containing Y. For simplicity assume that Y is an interior point of p.

(b) Find the sequence ξ of edges in J_p through which $\pi(Y)$ passes.

(c) Apply the planar unfolding procedure relative to ξ, as described above, to obtain the required shortest path $\pi(Y)$.

Concerning the complexity of the path-finding procedure just outlined, assume that the face f of K containing Y is specified too. Each such f is partitioned by Q into at most $O(n)$ subregions. It is then straightforward to locate in time $O(n)$ the subregion of f containing Y, and thus the peel p containing Y (for example, pass a straight line through Y and compute its intersections with $Q \cap f$, from which the region containing Y can be readily calculated). Given p, we obtain ξ in $O(n)$ time from the pre-calculated associated tree, and the planar unfolding procedure can then be applied to ξ also in time $O(n)$.

Corollary: After pre-processing, the shortest path from a fixed point X to any point $Y \in S$ can be computed in time $O(n)$, using a data-structure whose size is $O(n^2)$.

The main problem that we still face is that of computing Q. This will be done in the following section.

5. The Peels Construction Algorithm.

In this section we present an algorithm for partitioning the surface of K into peels, which runs in time $O(n^3 \log n)$. The algorithm constructs a tree T, called the *slice tree* as follows. Let us define a slice σ in terms of two "starting orientations" $\theta_1 < \theta_2$ and a "terminal face" f, which are assumed to have the following properties:

(i) Each of $p(\theta_1)$ and $p(\theta_2)$ reaches a vertex of K either before reaching f or on f itself.

(ii) For each $\theta \in (\theta_1, \theta_2)$ the path $p(\theta)$ reaches f and does not meet a vertex before its first exit from f.

(iii) For each $\theta \in (\theta_1, \theta_2)$ let $Z_\theta \in p(\theta)$ be the point at which this path leaves f (for the first time); then all the points Z_θ lie on the same edge of f (which we call the *terminal edge* of σ).

The corresponding slice, denoted as $\sigma(\theta_1, \theta_2, f)$, is defined to be the union of all paths $p(\theta, Z_\theta)$, for $\theta \in (\theta_1, \theta_2)$. Note that the set of points at which geodesic paths in $\sigma(\theta_1, \theta_2, f)$ enter f is also a subsegment of some edge of f. Note also that all the geodesic paths $p(\theta, Z_\theta)$ in this slice pass through the same sequence of edges of K, and that if σ is unfolded into the plane along this sequence of edges, its image is a triangle bounded by the two rays emerging from the origin (X) at orientations θ_1 and θ_2 and by the terminal edge of σ (cf. Fig. 5.1).

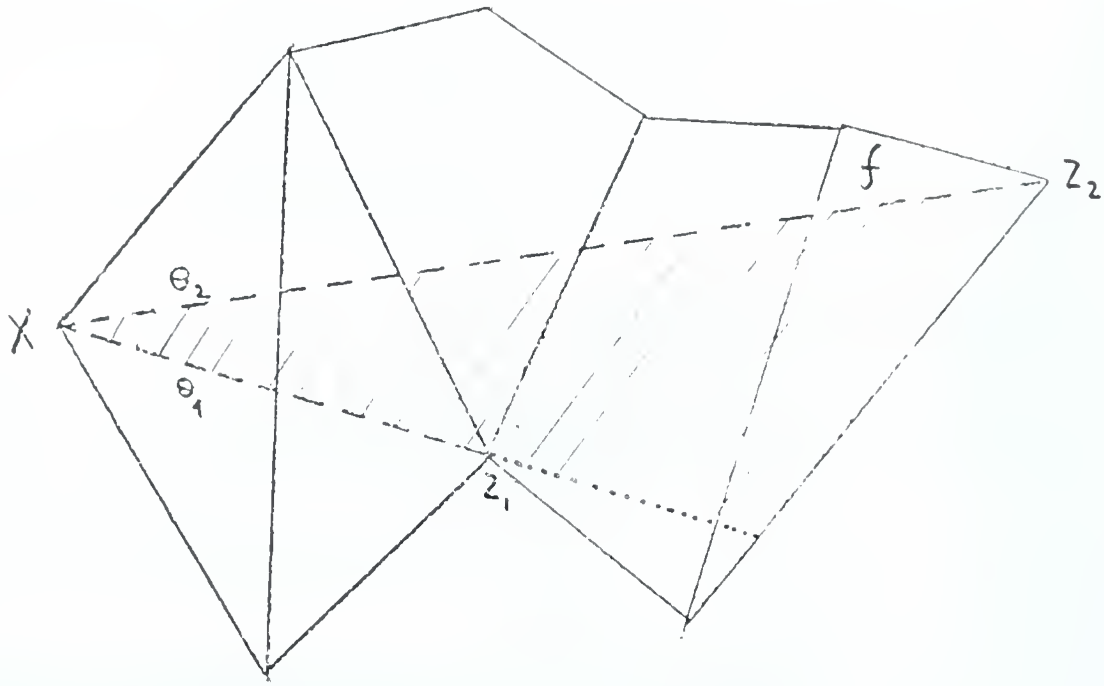


Fig. 5.1. Planar layout of a slice.

A slice is roughly meant to correspond to some portion of a peel, which is 'encoded' implicitly by the range of starting orientations and the terminal face f. The correspondence is such that for each 'true' slice $\sigma = \sigma(\theta_1,\theta_2,f)$ the sequence of edges that it meets is one of the sequences associated with some peel p (in the manner described in the preceding section), and the range (θ_1,θ_2) is contained in the corresponding range $I(p)$. Note that the above definition does not ensure that the geodesics $p(\theta)$ within σ for $\theta \in (\theta_1,\theta_2)$ are actually shortest paths. However, the algorithm to be described below will make sure that each slice σ that it will construct will correspond to some peel p in the manner just described, so that at least one geodesic path passing through the sequence of edges defined by σ is indeed a shortest path.

The slice tree that our algorithm will construct is defined as follows. Each node of T except for its root is a slice. The root is a dummy slice and its sons are the slices $\sigma(\theta_i,\theta_{i+1},f_0)$, $i=1, \ldots ,s$, where f_0 is the initial face of K containing X and where $\theta_1, \ldots ,\theta_s$ are the orientations of the segments connecting X to the vertices of f_0, ordered in angular order about X. Let $\sigma = \sigma(\theta_1,\theta_2,f)$ be a node of T. Its sons are obtained by extending σ one face past f. Specifically, let e be the terminal edge of σ, i.e. the edge of f containing all the terminal points

Z_θ for $\theta \in (\theta_1,\theta_2)$, and let f' be the face of K adjacent to f at e. If the geodesic paths in σ have already passed through f' (before reaching f) then σ is called a ***terminal*** slice, and remains a leaf of T. Otherwise we extend all paths $p(\theta)$ for $\theta \in (\theta_1,\theta_2)$ into f', and let τ denote the set of points at which these paths leave f'. Plainly τ is a connected portion of the boundary of f' and thus is a union of portions $\tau_1, \ldots, \tau_q$ of edges of f' (actually each τ_j for $1<j<q$ is a full edge). For each $1 \le j < q$ let $\psi_j \in (\theta_1,\theta_2)$ denote the orientation of the geodesic path in σ which reaches the vertex at which τ_j and τ_{j+1} meet. Put also $\psi_0 = \theta_1$ and $\psi_q = \theta_2$. Then slices that are candidates for the sons of σ in T are the slices $\sigma(\psi_{i-1},\psi_i,f')$, $i=1, \ldots, q$ (see Fig. 5.2).

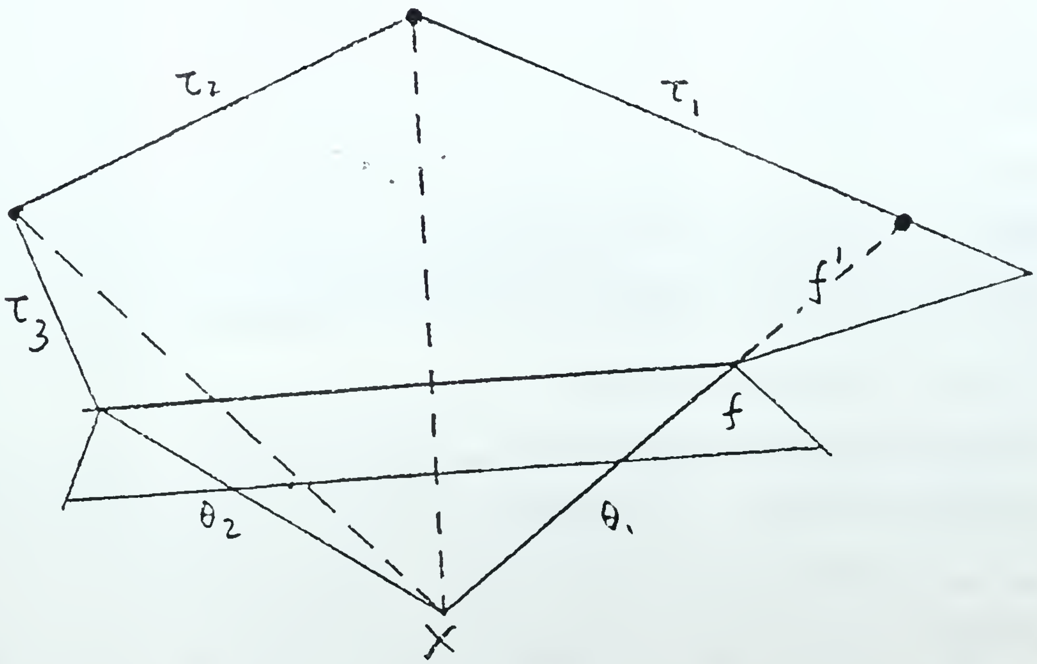


Fig. 5.2. Extending a slice past an edge.

The problem with extending the tree in this somewhat uncontrolled manner is that this might cause us to add to the tree slices for which the sequences of edges of K which they define may be such that no shortest path passes through it. Note that the number of ***true*** slices (i.e. slices for which there exists at least one shortest path passing through the sequence of edges which they define) is $O(n^2)$, in virtue of Proposition 4.8, but that we have no obvious way to estimate the total number of possible "false" slices that might be introduced by the construction in the preceding paragraph.

We therefore design our algorithm so that it always add only true slices to the tree T. More specifically, the algorithm will maintain the following two invariants:

(a) The tree constructed by the algorithm contains only true slices (in the above sense).

(b) With each slice $\sigma = \sigma(\theta_1,\theta_2,f)$ we associate a ***terminal portion*** σ_f which is a polygonal subregion of $\sigma \cap f$. The terminal portions of slices in the tree will be pairwise disjoint and each terminal slice portion will contain the set of all points Z in f for which the shortest path to Z within σ is shorter than than any path to Z contained within one of the slices present in T.

To maintain the invariant (b), the algorithm will have to store in the tree that it constructs additional information concerning the terminal portion of each slice. Terminal slice portions that are proper subsets of their corresponding slice portions $\sigma \cap f$ can arise due to the overlapping of such a $\sigma \cap f$ with other slices already present in the tree (and reaching the same terminal face). This implies that as the tree is being constructed, these trimmed portions may be trimmed again and again as new slices are added to the tree until they reach their final value at the end of the algorithm. We will show below that proper assembly of the terminal portions of the slices in the final tree gives us all the peels that we seek.

Each iteration step of the algorithm adds a new slice to the tree. Let T_k denote the slice-tree after the k-th iteration of the algorithm. For each starting orientation θ let Z_θ denote the farthest point on $p(\theta)$ such that $p(\theta,Z_\theta)$ is wholly contained within a slice of T_k. Define the *front* $F(T_k)$ of T_k to be the set of all points Z_θ which lie on edges of K. Let $W(T_k)$ denote a point $Z_\theta \in F(T_k)$ for which the length of $p(\theta,Z_\theta)$ is smallest, and let $m(T_k)$ denote this smallest length.

Initially, as defined above, T_0 consists of all the slices contained in the face f_0 containing X, all being children of a common dummy root.

At the $(k+1)$-st step, the algorithm picks $Z_\theta = W(T_k)$, and obtains the slice σ in T_k containing Z_θ. Suppose for the moment that Z_θ lies in just one such slice. Let f be the terminal face of σ, let e be the edge of K containing Z_θ, and let f' be the other face of K adjacent to f at

e.

Let U_θ denote the farthest point along $p(\theta) \cap f'$ (lying on another edge of f'), and let σ' be the slice which extends σ past e and which contains $p(\theta, U_\theta)$ (again we suppose for the moment that only one such slice exists). Then we have

Lemma 5.1: σ' is a true slice, in the sense that there exists at least one shortest path passing through the sequence of edges which σ' defines.

Proof: What we have to show is the existence of a starting orientation θ' within the range of starting orientations of σ' and a point $U \in p(\theta') \cap f'$ for which $p(\theta', U)$ is a shortest path to U. We take θ' to be θ, and U to be a point in $p(\theta) \cap f'$ sufficiently near Z_θ. Suppose to the contrary that $p(\theta, U)$ is not one of the shortest paths to U. Let ψ be the starting orientation of the shortest path to U. Two cases can arise:

(i) The point U belongs to some slice in T_k. But if this is the case for all points U sufficiently close to Z_θ, then Z_θ must be a ridge point and lie on the boundary of more than one slice in T_k, contrary to assumption.

(ii) Hence U does not belong to any slice in T_k, so that $p(\psi)$ must reach the front $F(T_k)$ at some point V before reaching U. But then the length of $p(\psi, V)$ is at least that of $p(\theta, Z)$, so that if U is chosen sufficiently near Z_θ the path $p(\psi, U)$ cannot be a shortest path to U. (To show this note that the additional length of $p(\theta)$ between Z_θ and U will be strictly smaller than the length of $p(\psi)$ between V and U if one chooses U sufficiently near Z_θ, for otherwise an impossible situation as in case (i) above would occur.) This proves the lemma. Q.E.D.

The cases in which Z_θ belongs to two slices in T_k, or in which there are two slices which extend σ and contain $p(\theta, U_\theta)$ deserves a slightly modified treatment. It can be shown that, except for some rare cases which can be detected by a purely local analysis of the structure of geodesic paths near Z_θ, one can extend either of the slices containing Z_θ in the first case, or use any of the extended slices obtained in the second case, and Lemma 5.1 will still hold for this extended new slice, although its proof will have to be somewhat modified.

The algorithm will then add the slice σ' as a son of σ to the slice tree, thus maintaining property (a). To maintain (b) the algorithm will have to trim the terminal portion of σ' to keep it disjoint from the terminal portions of other slices reaching the same face, and possibly also trim these other terminal portions.

To understand how such a trimming is to be done, suppose first that we are given just two slices, both reaching the same terminal face f and overlapping one another on f. Then the following basic procedure is applicable.

Slice Trimming Procedure: Let $\sigma_1 = \sigma(\theta_1, \theta'_1, f)$, $\sigma_2 = \sigma(\theta_2, \theta'_2, f)$ be two slices with the same terminal face f. Apply the planar unfolding procedure to σ, σ' to obtain planar triangular layouts of these two slices. Move these two layouts in the plane so that the two copies of the face f in these layouts coincide with the standard plane representation of f. Let X_{σ_1}, X_{σ_2} be the positions of the point X in these two new layouts, and let l be the perpendicular bisector of $X_{\sigma_1}X_{\sigma_2}$. Let f_{12} (resp. f_{21}) denote the portion of f lying on the X_{σ_1}-side (resp. the X_{σ_2}-side) of l. We then replace $\sigma_1 \cap f$ (resp. $\sigma_2 \cap f$) by $\sigma_1 \cap f - \sigma_1 \cap \sigma_2 \cap f_{21}$ (resp. $\sigma_2 \cap f - \sigma_1 \cap \sigma_2 \cap f_{12}$). See Fig. 5.3.

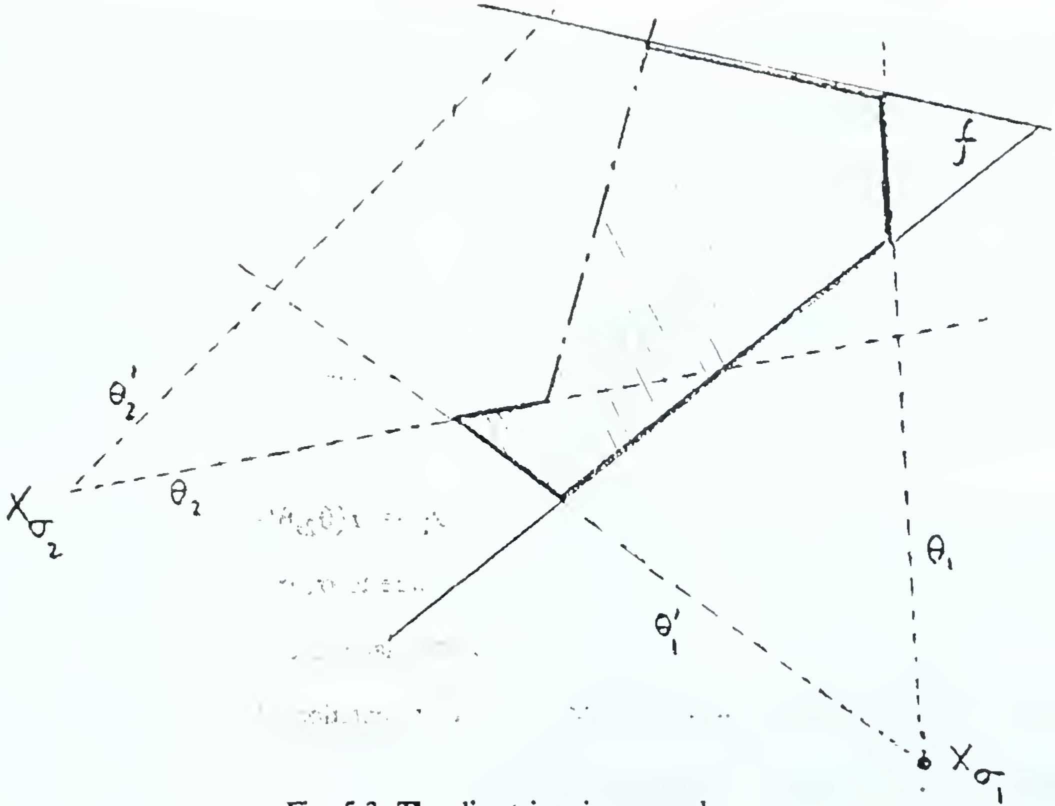


Fig. 5.3. The slice trimming procedure.

In other words, for points Z which can be reached by two distinct geodesic paths, lying in two different slices, the slice trimming procedure determines which of these two paths is shorter, and removes the point Z from the other slice.

Let us consider in more detail the computational aspects of this procedure. As the slice tree is being built, we can store with each slice σ the position X_σ of the point X in the planar layout of σ in which f lies in its standard plane position (X_σ will be represented by quantities a_σ, θ_σ as in the preceding section, which can be easily updated as we extend σ to an adjacent face).

Given two slices $\sigma_1 = \sigma(\theta_1, \theta'_1, f)$ and $\sigma_2 = \sigma(\theta_2, \theta'_2, f)$ meeting at the same terminal face f, identify σ_j with its specific planar unfolding used in the slice trimming procedure above, $j = 1,2$. Then it is easily seen (cf. Fig. 5.3) that $\sigma_1 - \sigma_1 \cap \sigma_2 \cap f_{21}$ is a bounded polygonal region which is star shaped with respect to X_{σ_1}, and which is bounded by at most six segments, including the two rays emanating from X_{σ_1}. We regard the boundary of this region, excluding the two rays from X_{σ_1}, as the graph of a function expressing the distance ρ from X_{σ_1} as a function of the orientation $\theta \in (\theta_1, \theta'_1)$ and write it as $\rho_{\sigma_1, \sigma_2}(\theta)$.

Now suppose that at a certain time during its construction, the tree T contains slices $\sigma_1, \ldots, \sigma_t$ all reaching the same terminal face f. Then to enforce property (b), the terminal portion of each σ_i will have to be replaced by $\nu(\sigma_i) \cap f$, where

$$(*) \qquad \nu(\sigma_i) = \sigma_i - \bigcup_{j \neq i} \sigma_i \cap \sigma_j \cap f_{ji}$$

Note that $\nu(\sigma_i)$ is a star-shaped region (with respect to X_{σ_i}) whose boundary, except for the two rays emanating from X_{σ_i}, is represented by the function

$$\rho_{\sigma_i}(\theta) = \min_{j \neq i} \rho_{\sigma_i, \sigma_j}(\theta) \ , \qquad \theta \in (\theta_i, \theta'_i)$$

In other words, ρ_{σ_i} is the "lower envelope" of $t-1$ polygonal lines, each consisting of at most four segments, as these lines are viewed from the point X_{σ_i}.

We claim that the following property holds:

Lemma 5.2: If ρ_{σ_i} consists of t_i segments, then $\sum_i t_i = O(t)$.

Proof: To prove this claim, we proceed in a manner quite similar to the estimate of the size of Voronoi diagrams used in [Sh]. That is, we define an undirected graph G whose nodes are the points X_{σ_i}, and in which an edge connects X_{σ_i} to X_{σ_j} if the trimmed portions of σ_i and σ_j have an edge in common, and if this edge is not contained in any of the four rays emanating from X_{σ_i} and X_{σ_j}. It is easy to show that G is a planar graph by mapping each edge $(X_{\sigma_i}, X_{\sigma_j})$ of G to the union of two segments connecting X_{σ_i} and X_{σ_j} to a point on the corresponding common edge. Moreover, the graph G has no multiple edges, a fact which can be established as in the proof of Lemma 4.5. It therefore follows by Eular's formula that G has $O(t)$ edges. Finally, each of the edges obtained by the trimming process which is not recorded in G lies on a ray bounding some slice σ_i, and plainly there can be at most $O(t)$ such edges. These observations establish our claim. Q.E.D.

Now by the results of the preceding section, $t \leq n$, since the total number of "true" slices reaching the same face is at most n. This means that the total number of edges separating trimmed terminal portions of slices within a given face is at most $O(n)$ during each stage of the

algorithm. Moreover, as we add a new slice σ to the tree we need to update the functions ρ_{σ_i} for all slices σ_i reaching the same terminal face f as σ. Suppose as before that ρ_{σ_i} consists of t_i segments. Then its updated value due to the appearance of σ can be computed in a straightforward manner using (*) in time $O(t_i)$, so that updating of all these functions can be accomplished in time $O(n)$. Moreover, each new edge added to any of these functions, and only such edges, will also appear in the graph of the new function ρ_σ, and it is an easy matter to assemble all these edges and thus obtain ρ_σ in time $O(n)$.

Remark: Lemma 4.6 implies that the final trimmed value of a slice is convex (when properly unfolded). The procedure sketched above might however temporarily lead to nonconvex trimmed slices. We do not know whether nonconvex intermediate slices can actually be obtained.

We therefore conclude that one can maintain property (b) in time $O(n)$ per each step of the algorithm.

The algorithm also needs to update the values of $W(T_k)$ and $m(T_k)$ in view of the addition of σ' to the tree. Note that the new front of the tree is obtained from the previous front by deletion of the "entering" segment of σ', by addition of the terminal edge of σ', and by possibly shortening other edges of the front due to the trimming procedure described above. Note that the deletion of the entering edge of σ' will in general have split (a portion of) the terminal edge of σ into two subsegments which still belong to the front, and that similar splits or replacements of edges will result from the trimming process. This suggests that we represent the front of the tree as a union of subsegments of edges of K. For each segment e in the front we can easily compute the point $W(e)$ on e for which the geodesic path to $W(e)$ through the slice bounded by e is shorter than that to any other point on e, and also the length $m(e)$ of this path. We then maintain a priority queue containing all the relevant segments e constituting the front, ordered by the value of $m(e)$. At each addition of a new slice σ' to the tree we delete from the priority queue the terminal segment of the ancestor σ of σ', add back the two subsegments of this segment still in the front, and add the terminal edge of σ'. Since the trimming step can result in a new terminal slice portion having no terminal edge, and since the terminal edges of existing slices

may also be trimmed by that procedure, the priority queue of front edges will have to be updated in an appropriate manner. Since the incremental trimming procedure described above will produce at each step only $O(n)$ slice-bounding segments, it follows that at most $O(n)$ updates of the priority queue will be required at each stage. Once these updates are performed, the updated values $W(T_k)$ and $m(T_k)$ are easily available and can be retrieved in the next iteration of the algorithm.

The algorithm terminates when the priority queue representing the front of the tree becomes empty, i.e. when the front itself becomes empty.

When this happens we still need a final phase that will construct the peels of K from the slice tree. To this end let $\sigma = \sigma(\theta_1, \theta_2, f)$ be a leaf of T. The final trimmed value of σ is defined as

$$\mu(\sigma) = \bigcup_{\sigma'} \sigma \cap \tau(\sigma')$$

where σ' ranges over all slices on the path in T to σ, and where $\tau(\sigma')$ denotes the final trimmed terminal portion of σ'. That is, we collect all trimmed terminal portions of slices lying along the path to σ in T, but restrict each such portion to the wedge between the two starting orientations defining σ.

Note that $\mu(\sigma)$ need not be a full peel. In fact, a necessary and sufficient condition for $\mu(\sigma)$ to be a peel is that the two bounding geodesics $p(\theta_1)$ and $p(\theta_2)$ are not trimmed, and still reach vertices of K within (the closure of) $\mu(\sigma)$. If, say, $p(\theta_1)$ has been trimmed, let σ_1 be the slice whose range of starting orientations is adjacent to that of σ at θ_1. Then it is easy to see that the peel containing $\mu(\sigma)$ also contains $\mu(\sigma_1)$. These observations makes it clear that we can construct all peels by a depth-first traversal of T, visiting sons of a slice in, say, counterclockwise order of their ranges of starting orientations.

To estimate the time required by the algorithm we note that the maintenance of property (b) is the costliest part of the algorithm, in which the updating of the priority queue representing the front of T may require $O(n \log n)$ time for each step (the trimming procedure itself requiring

only $O(n)$ time). Since the algorithm adds at most $O(n^2)$ slices to the tree, it follows that the algorithm will run in time $O(n^3 \log n)$.

The correctness of the algorithm follows from the following considerations. First note that the peels as constructed by the algorithm are pairwise disjoint by construction. We also claim that they cover the whole surface of K. To show this, it suffices to prove that every true slice is constructed by the algorithm. Indeed, if this latter property is known to hold, then for each point Z on the surface of K let $p(\theta, Z_\theta)$ be a shortest path to Z. The sequence of edges and faces through which this path passes defines a true slice σ which will appear in the final tree T. It is easy to verify that Z will belong to the final trimmed terminal portion of σ.

To see that all true slices are constructed by the algorithm, we note that once the algorithm has added a slice to the tree with a nonempty terminal edge which connects that slice to another true slice, then this edge will not be wholly deleted by the trimming procedure, and eventually the following slice will also be picked up by the algorithm and added to the tree. This implies in a straightforward inductive manner that all true slices are added to the tree.

Hence the peels produced by the algorithm are pairwise disjoint and cover the whole surface of K. It is now a simple matter to prove, arguing as in the proof of Lemma 5.1, that each such peel is one of the peels defined in the preceding section, and thus to conclude the proof of correctness of the algorithm.

In view of the discussion at the preceding section, we thus have the following summary theorem.

Theorem 5.2: Given a convex polyhedron K with n vertices and a point X on its surface, one can preprocess K by a procedure which runs in $O(n^3 \log n)$ time. This procedure produces a data structure of size $O(n^2)$, with the aid of which one can find in $O(n)$ time the shortest path along the surface of K from X to any other specified point.

Remark: It seems quite likely that the algorithm developed in this section is not optimal, as it requires $O(n^3 \log n)$ time to construct a quadratic data structure (and then search the structure in only linear time to find a shortest path). After the original submission of this paper, Mount [Mo]

has recently proposed an improved approach, in which the data structure only maintains points of intersections of slices with the edges of K (rather than with its faces, as done here). Thus his data structure is a collection of disjoint intervals on the edges of K, rather than a collection of disjoint polygons on the faces of K, making it much easier to maintain this structure, and thereby reducing the running time to $O(n^2 \log n)$.

References

[AHU]

A. Aho, J. Hopcroft and J. Ullman, *The Design and Analysis of Computer Algorithms*, Addison Wesley, Mass. 1974.

[Al] A.D. Alexandrov, *Konvexe Polyeder* (translated from Russian), Akademie-Verlag, Berlin 1958.

[Ch] B. Chazelle, A theorem on polygon cutting with applications, *Proc. 23rd IEEE Symp. on Foundation of Computer Science*, 1982, pp. 339-349.

[Co] G.E. Collins, Quantifier elimination for real closed fields by cylindrical algebraic decomposition, in *Second GI Conference on Automata Theory and Formal Languages*, Lecture Notes in Computer Science, Vol. 33, pp. 134-183, Springer Verlag, Berlin.

[FAV] W.R. Franklin, V. Akman and C. Verrilli, Voronoi diagrams with barriers and on polyhedra, Tech. Rept., Electrical, Computer and Systems Engineering Dept., Rensselaer Polytechnic Institute, Troy, N.Y., November 1983.

[FA] W.R. Franklin, and V. Akman, Minimal paths between two points on/around convex polyhedra, Tech. Rept., Electrical, Computer and Systems Engineering Dept., Rensselaer Polytechnic Institute, Troy, N.Y., May 1984.

[Ga] M. Gardner, *The 2nd Scientific American Book of Mathematical Puzzles and Diversions*, Simon and Schuster, N.Y. 1961.

[LP] D.T. Lee and F.P. Preparata, Euclidean shortest paths in the presence of rectilinear barriers,

Networks 14 (1984), pp. 393-410.

[LW] T. Lozano-Perez and M.A. Wesley, An algorithm for planning collision-free paths among polyhedral obstacles, *Comm. ACM* 22 (1979) pp. 560-570.

[Mo] D.M. Mount, On finding shortest paths on convex polyhedra, Tech. Rept., Computer Science Dept., University of Maryland, October 1984.

[NP] J. Nievergelt and F.P. Preparata, Plane sweeping algorithms for intersecting geometric figures, *Comm. ACM* 25 (1982), pp. 739-747.

[OSB] J. O'Rourke, S. Suri and H. Booth, Shortest paths on polyhedral surfaces, extended abstract, Dept. of Electrical Engineering and Computer Science, Johns Hopkins University, September 1984.

[Pa] C.H. Papadimitriou, An algorithm for shortest-path motion in three dimensions, manuscript, Computer Science Department, Stanford University, July 1984.

[Re] J. Reif, Minimum $s-t$ cut of a planar undirected network in $O(n \log^2 n)$ time, *SIAM J. Computing* 12 (1983) pp. 71-81.

[SS] J.T. Schwartz and M. Sharir, On the piano movers' problem: II. General techniques for computing topological properties of real algebraic manifolds, *Adv. in Appl. Math.* 4(1983), pp. 298-351.

[Sh] M.I. Shamos, *Computational Geometry*, Ph.D. Dissertation, Yale University, 1975.

[To] M. Tompa, An Optimal Solution to a Wire Routing Problem, *J. Comp. Syst. Sciences* 23(1981), pp. 127-150.

[Wa] B.L. Van der Waerden, Algebra, 5th Edition, Springer Verlag, Berlin, 1960.

c.1

NYU CompSci TR-138
Sharir, Micha c.1

On shortest paths in
polyhedral spaces

DATE DUE	BORROWER'S NAME